Nahom Worku
ID: 218776070

EECS 2021 - Major Project

INTRODUCTION:

This is a project where a DHT11 Temperature-Humidity sensor is integrated with ATMEGA328p microcontroller (grove board) to detect and display the relative humidity in a room. What is a DHT11? When we talk about humidity, we usually talk about relative humidity (RH). The relative humidity is expressed as a percentage, or ratio, of the amount of water vapor present in the air compared to the maximum amount of water vapor that could be present at that temperature. The value of the humidity is displayed as binary numbers through a one-digit-7-segment display and an LED. The ATMEGA328p microcontroller on the grove board is connected to MPLAB SNAP - PG164100 and programmed in C and Assembly language.

CONTEXT:

This project is a project that has a complete program by using the ATMEGA328p microcontroller's internal hardware. I went through every process in the ATMEGA328p datasheet and implemented it accordingly. The ATMEGA328p microcontroller has the ability to connect to MPLAB SNAP - PG164100 via digital communication. The three I/O registers, DDRx, PORTx, and PINx, of the ATMEGA328p, which is an 8-bit AVR microcontroller, are used to set OUTPUTS, INPUTS, receive and send data to ports and pins.

In this project, we use some of the input functions from C and assembly language The ATMEGA328p microcontroller receives data from the DHT11 sensor and stores the data in registers. Each bit of data sent from the sensor is collected by the microcontroller by reiterating 8 times for 8-bits data (relative humidity).

TECHNICAL REQUIREMENTS / SPECIFICATIONS:

As mentioned above, this project uses a DHT11 sensor to display the humidity (in %) on a one-digit-7-segment display and 8 LEDs.

To determine if two consecutive bits are ON (1) or OFF(0), an AND gate IC chip (74LS08). This helps to determine whether the humidity is too high or not. For example, if the two bits (from the most significant bit) are 1, that means the humidity is at least 192%. If that is the case an active buzzer is triggered although it rarely happens. The humidity being greater than 100% means there is a supersaturation condition: “In the absence of a foreign body on which droplets or crystals can nucleate, the relative humidity can exceed 100%, in which case the air is said to be supersaturated”[2].

The data format transmitted between the DHT11 sensor and microcontroller is 40 bits data.

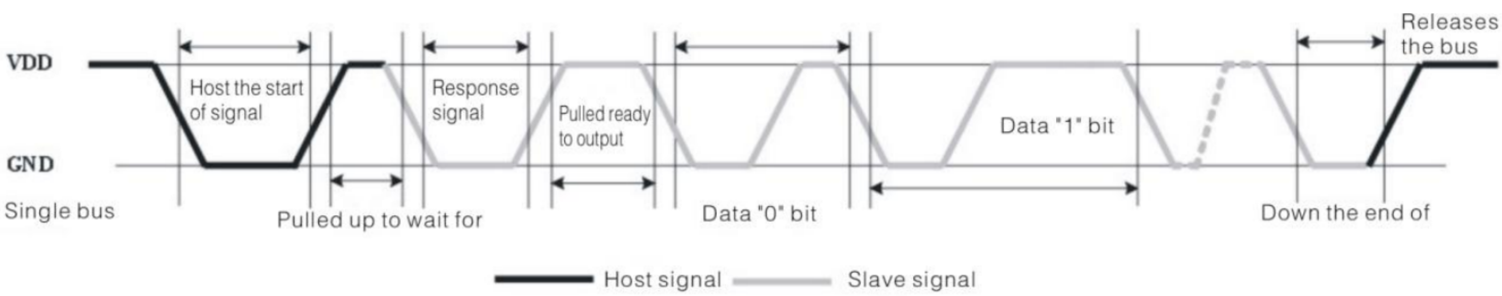
The data format look like the following:

- 8 bit humidity integer data + 8 bit the Humidity decimal data +8 bit temperature integer data + 8 bit fractional temperature data + 8 bit parity bit [1]

Since this project is only aiming to get the humidity data, the first 8 bit integer data is collected (extracted). The decimal date of the Humidity can also be collected but, for simplicity this project does not include that. The bits humidity integer data received by the MCU is stored in a register and shifted to the left shift register (LSB) one after the other. This is done through conditional loops

For example, if the first 8 bits of data received from the DHT11 is 0b00100100, it means the relative humidity (RH) is 36%.

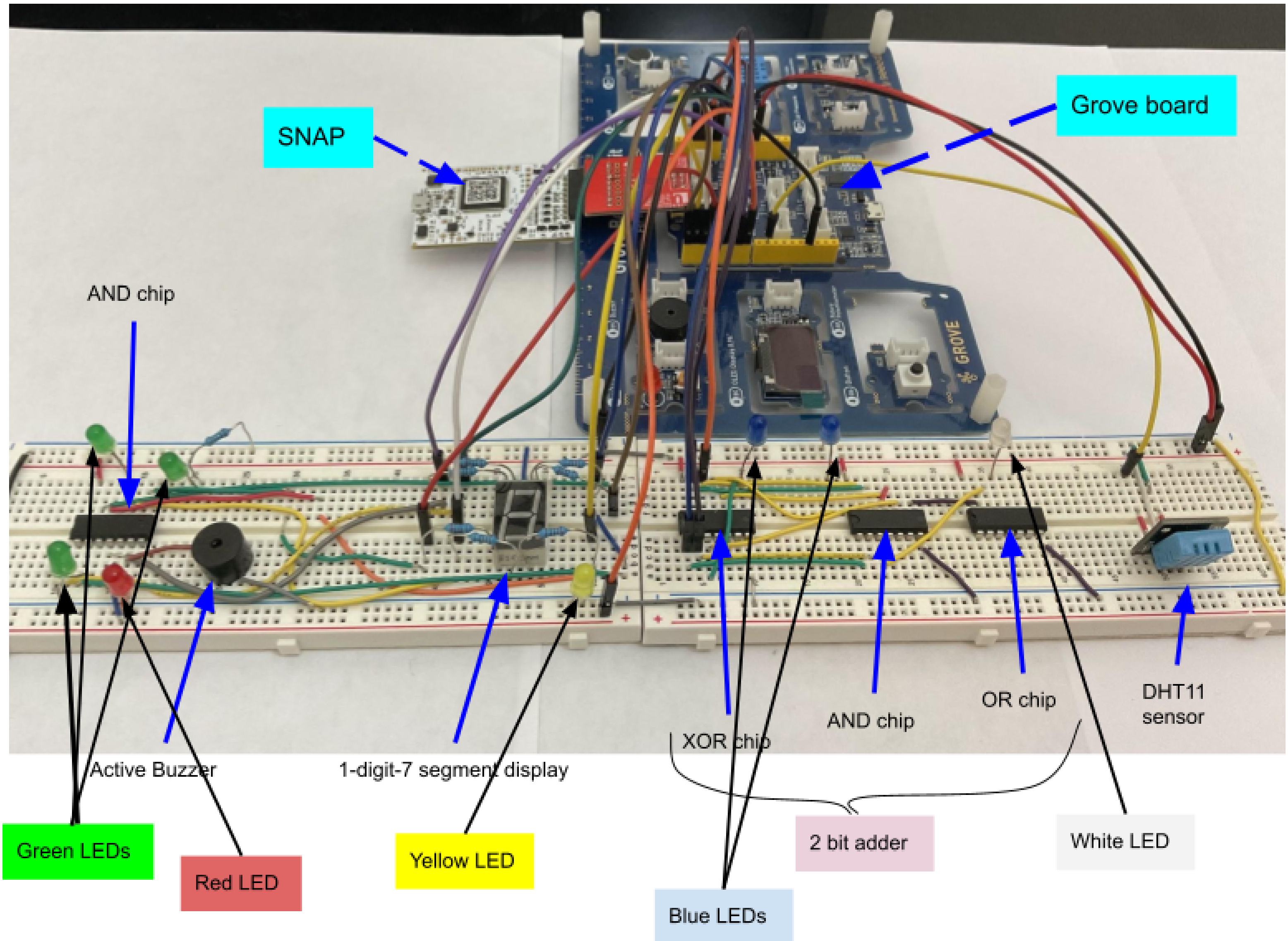
How does the DHT11 measure temperature and humidity?



Data Timing Diagram [1]

The DHT11 sensor communicates with the ATMEGA328p microcontroller through 4 steps. The 4 steps are explained in more detail in the datasheet. To summarize, to receive data from the DHT11 sensor, all that’s needed is to send low and high pulses with certain delays in between. There are 2 types of data bit format: bit data “0” format and bit data “1” format. The difference between these two types of format is the length of durations of high and low pulses [1].

Now we know how the DHT11 sensor and the ATMEGA328p communicate, let’s see how they are wired.



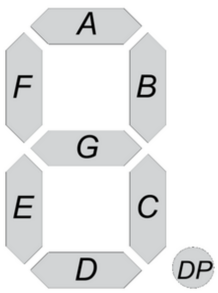
- ➢ Green LEDs indicators when two consecutive bits are 1 or 0.
- ➢ Red LED signals whether the last 2 bits are ON or OFF.
- ➢ Yellow LED part of the 7 segment display. Displays the most significant bit of the humidity.
- ➢ Blue LEDs used to display the sum of two 2-bit binary numbers.
- ➢ While LED used to display the carry from the 2-bit adder

The value of the humidity gathered by the DHT11 is sent to PORTD and PORTC.

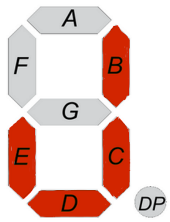
The 7 segment display is connected to PORTD. A 2-bit binary adder is built with the inputs coming from PORTC. Since the 2-bit adder has 4 inputs, PC0, PC1, PC2, PC3 are used as the inputs. The first input is from PC0, PC1 and, second input

PC2, PC3 The 2-bit adder is built using XOR, AND, and OR gate IC chips. It includes one half and full adder. This 2-bit adder is built to demonstrate that we can build a circuit by using the data gathered from the DHT11 sensor. Although the 2-bit adder is a fairly simple circuit, it has many applications. For building more complicated circuits, a Karnaugh map and Boolean Algebra can be used to generate desired outputs.

One-digit-7 segment display connection:



- B is connected to PD0,
- C is connected to PD1,
- D is connected to PD2,
- E is connected to PD3,
- F is connected to PD4,
- A is connected to PD5,
- G is connected to PD6,
- An LED (yellow) is connected to PD7,



The 7 segment display is used to output the binary humidity value. For example, if the binary value of the relative humidity is 0b00001111, B, C, D , and E would all be turned ON as shown in the figure.

As it is known, a one-digit-7 segment display can display up to 15 (in base 10) in hexadecimal and up to 9 as base 10 number. To do this a 7447 Binary Coded Decimal (BCD) chip can be used. This is another example that indicates this project can be improved in various ways.

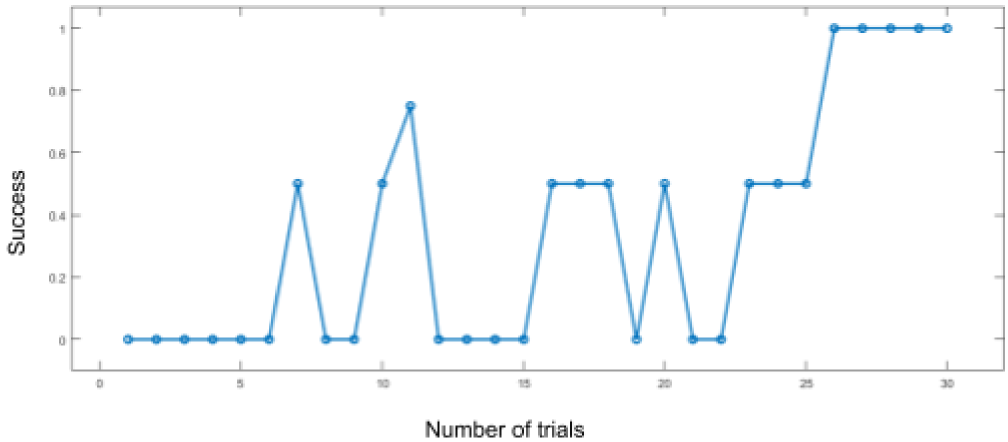
COMPONENTS LIST:

- Arduino Grove Board
- MPLAB SNAP - PG164100 (EECS 2021 Course Kit)
- DHT11 Temperature and Humidity sensor
- One-Digit-7-segment display
- 2 Quad 2-in AND (74LS08)
- OR chip (74LS32)
- XOR chip (74LS86)
- Active buzzer
- LEDs
- DuPont Wires
- Breadboards
- Resistors

PROCEDURE:

First, the DH11 sensor is connected to ground, VCC and digital pin 9 of the Arduino grove board. Then the 7 segment display is connected to PORTD (PD 0-6). Since the output is an 8 bit data and the 7 segment display only has 7 LEDS, the most significant bit (PD7) is connected to an external LED. This way, the 8-bit data sent from the DHT11, is displayed in binary through the 7 segment display plus the LED. An AND IC chip 74LS08 is used to build a circuit that indicates whether two consecutive bits are 1 or 0 starting from the least significant bit. If 2 consecutive bits are 1, a green led would be turned ON (since 1 & 1 = 1) and OFF if the bits are 0. A 2-bit adder is built using an AND (74LS08), OR chip (74LS32) and XOR chip (74LS86) to add two 2-bit numbers and display their sum and carry using LEDs. The inputs for this 2 bit adder are connected to PORTC (PC 0-3). Power up the circuit by connecting the grove board and SNAP.

TEST:



In this number of trails vs success graph where the number of trials is graphed in the x axis and the success in the y axis. This project was not an easy project. For the first 6 trials, nothing was working hence 0 is given for the success in the graph. That was because of the code. I didn’t include the correct delays needed for the DHT11 sensor to initialize its readings. After fixing that problem, the sensor was giving out signals but the binary values were way too high to make sense. As a result, for the 7th trial 0.5 was given for the success axis. After that, the trails were up and down for problems such as wiring, assigning the correct PORT etc. After the 23rd trial, the system started to work although it was not functioning with 100% accuracy. For this reason, trials 23, 24, and 25 are given 0.5 in the success axis. After the 25th trial, after a couple of days of careful testing, I found out the DHT11 temperature sensor was damaged. After buying a new DHT11 sensor, the system fully worked for the first time in the 26th trial. The result was consistent for the rest of the trials.

CONTINGENCY

My initial plan for this project was to display the relative humidity value on an LCD display and 4-digit-7 segment display.

This requires a deep understanding and knowledge of C and assembly language. Since I don't have the required level of knowledge of C and assembly, changing my initial plan was necessary. In addition, the ATMEGA328p MCU only has one port (PORTD) that has 8 digital pins (D0 - D7). This makes it harder to integrate a 4-digit-7 segment display and LCD. As a result, using other microcontrollers with more pins is necessary. For this reason, using the PIC16F777 microcontroller was part of my plan. As I was doing my research, I found out the PIC16F777 microcontroller is not supported by the SNAP. Since I started planning for this major project towards the start of the term, changing my plan was not a problem. After doing more research, using the grove board with the SNAP became the most convenient choice.

As mentioned earlier, to represent a binary number in base 10, one can use the 7447 Binary Coded Decimal (BCD) along with a 7-segment display. Using this BCD was also part of my initial plan. However, I was not able to buy this chip as the chip is relatively expensive and would exceed my project budget. In addition, the delivery of ordering this chip would take so long as it is imported from elsewhere.

In my upper years of academic journey at Lassonde, I really hope I can expand on AVR microprocessors, C and assembly language programming as it would be beneficial for the co-op positions I want to acquire.

ADDITIONAL MATERIAL

Since the outbreak of COVID-19, scientists have done several research to help us defeat this virus. Among the research, monitoring CO2 levels, humidity, and other air quality related measures are suggested by these researchers. As a result people have been carrying CO2 level detectors to measure CO2 levels indoors, airplanes and many other places.

Humidity also plays an important role in keeping the viruses from spreading. According to studies, keeping the humidity of a room between 40 - 60% can slow the spread of COVID-19 [3][4]. This is a huge opportunity for individuals to take care of their chance of getting infected by COVID-19 by simply adjusting their room humidity. To adjust the humidity level, one can integrate a humidity controller in this project (i.e., using PORTC or PORTD)

CONCLUSION

As people are getting more studies and information about how they can keep themselves protected from covid, this project is built with these studies in mind. Studies have shown that keeping humidity between 40 to 60% can decrease the spread of coronavirus [3][4]. This project uses DHT11 which is a temperature and humidity sensor and other components such as 7 segment display, IC chips, buzzer, and leds. In this project the humidity of a room is gathered from the sensor and displayed on the 7 segment display as a binary number. An AND chip is used to determine whether 2 consecutive bits are 1 or 0. This helps determine if the humidity is above a certain % value. IC chips are used to make a 2-bit adder using the first 4 bits of the humidity values as inputs

Citation:

1. 秋月電子通商 トップページ - 電子部品・半導体【通販・販売】. https://akizukidenshi.com/download/ds/aosong/DHT11.pdf.

2. “Humidity.” *Wikipedia*, Wikimedia Foundation, 29 Nov. 2022, https://en.wikipedia.org/wiki/Humidity.

3. Jennifer Chu | MIT News Office. “Keeping Indoor Humidity Levels at a ‘Sweet Spot’ May Reduce Spread of Covid-19.” *MIT News | Massachusetts Institute of Technology*, https://news.mit.edu/2022/covid-humidity-spread-1116.

4. “Op-Ed: Humidity Can Aid in the Fight against Covid-19.” *News*, 3 Dec. 2020, https://www.hsph.harvard.edu/news/hsph-in-the-news/op-ed-humidity-can-aid-in-the-fight-against-covid-19/.

Textbook(s) used:

https://ww1.microchip.com/downloads/en/DeviceDoc/AVR-Instruction-Set-Manual-DS40002198A.pdf

https://electrovolt.ir/wp-content/uploads/2017/02/AVR_Microcontroller_and_Embedded_Electrovolt.ir_.pdf

Note: This project is made with the help of the textbooks mentioned above.